6. What is claimed is:

1. A tamper-resistant processing method comprising the steps of:

(1) storing a secret key index x corresponding to a public key of RSA (e, N; modulus N being a product of 2 primes p and q) in a storage device;

(2) inputting a ciphertext Y through an input means;

(3) calculating yp, a remainder of y, based on a modulus of either P or its multiple and yq, a remainder of y, based on a modulus of either Q or its multiple; and

(4) when calculating Cp which is a remainder of yp^xp based on a modulus of either of p or its multiple and calculating Cq which is a remainder of yq^xq based on a modulus of either q or its multiple, where a remainder of x based on a modulus of either of p-1 or its multiple is put as xp, and a remainder of x based on a modulus of either q-1 or its multiple is put as xq,

(4a) deciding which process (4b) or (4c) is to be executed for each processing of a bit block which is a bit string of at least 1 bit composing xp, xq;

(4b) executing a predetermined modular exponentiation calculation on said bit block to be processed by xp and for storing the calculation result in

the storage device;

(4c) executing a predetermined modular exponentiation calculation on said bit block to be processed by xq and for storing the calculation result in the storage device;

(5) calculating RSA decryption calculation, y^x mod N based on a difference between Cp and Cq, when the calculation of Cp about the whole of xp, and the calculation of Cq about the whole of xq are finished; and

(6) outputting the result of said RSA decryption calculation.

2. A tamper-resistant processing method of claim 1 wherein for said yp, yq, xp and xq, calculation be made as: yp = y mod p, yq = y mod q, xp = x mod (p-1), xq = x mod (q-1).

3. A tamper-resistant processing method of claim 1 wherein which one of said steps (4b) and (4c) is to be processed is determined with the use of a generated random number.

4. A tamper-resistant processing method of claim 1 wherein the process of said step (4a) is applied to a part of bit patterns of said xp or xq, and for a remaining part of the bit patterns, after said either one of step (4b) or (4c) is processed, another one is processed.

5. A tamper-resistant processing method comprising

the steps of:

(1) deciding which step is to be selected out of the following steps (2) and (3) for each processing of one operation unit;

(2) after transferring one operation unit in the bit pattern of data A in a memory to a first register R1, transferring one operation unit in the bit pattern of data B in the memory to a second register R2;

(3) after transferring one operation unit in the bit pattern of said data B to said second register R2, transferring one operation unit in the bit pattern in said data A to said first register R1;

(4) executing a predetermined arithmetic operation on the contents of said first register R1 and the contents of said second register R2;

(5) storing the result of said arithmetic operation in the memory,

(6) repeating the steps from (1) through (5) until said arithmetic operation for said data A and said data B is finished.

6. A tamper-resistant processing method comprising the steps of:

(1) deciding which step is to be selected out of the following steps (2) and (3) for each processing of one operation unit;

(2) after transferring one operation unit of data A in a memory to a first register R1, transferring one operation unit of data B in the memory to a second register R2;

(3) after transferring said one operation unit of the data A to said second register R2, transferring said one operation unit of the data B to said first register R1;

(4) executing a predetermined arithmetic operation on the contents of said first register R1 and on the contents of said second register R2;

(5) storing the result of said arithmetic operation in the memory;

(6) repeating the steps from (1) through (5) until said arithmetic operation on said data A and said data B is finished.

7. A tamper-resistant processing method of claim 6 wherein which one out of said steps (2) and (3) is to be processed is determined with the use of a generated random number.

8. A tamper-resistant processing method of claim 6 wherein said predetermined arithmetic operation is the operation for an arithmetic sum.

9. A tamper-resistant processing method of claim 6 wherein said predetermined arithmetic operation is the

operation for an arithmetic product.

10. A tamper-resistant processing method of claim 6 wherein said predetermined arithmetic operation is any one of the logical sum OR, logical product AND, and exclusive logical sum EXOR.

11. A tamper-resistant processing method comprising the steps of:

(1) selecting any one of unprocessed one operation unit in a bit pattern of data A in a memory;

(2) transferring said one operation unit of said data A selected to a first register R1;

(3) transferring one operation unit in a bit pattern of data B in the memory corresponding to said one operation unit of said data A selected to a second register R2;

(4) executing a predetermined arithmetic operation for the contents of said first register R1 and the contents of said second resister R2;

(5) storing a result of said arithmetic operation in the memory;

(6) repeating the steps from (1) through (5) until said arithmetic operation is finished on said data A and said data B.

12. A tamper-resistant processing method of claim 11 wherein corresponding to a generated random number,

said unprocessed one operation unit is selected.

13. A tamper-resistant processing method of claim 11 wherein said predetermined arithmetic operation is any one of logical sum OR, logical product AND, and exclusive logical sum EXOR.